

# 분산게임서버와 공간분할기법

숙명여자대학교 유석종

## 1. 서 론

최근 국산 온라인 게임이 국내외에서 각광을 받으며 한국이 온라인게임 기술을 주도하는 국가로 도약하고 있다. 전세계적으로 온라인 게임의 시장규모는 이미 PC게임, 비디오게임을 능가하고 있으며 이러한 온라인 게임의 성장 추세는 지속될 전망이다. 온라인게임의 성공요인을 몇가지로 분석할 수 있다. 첫째, 기존의 stand-alone형 게임에서 사용자는 인공지능 프로그램이 만들어낸 게임 패턴에 쉽게 적응하게 되는 반면, 온라인 게임은 다수의 참여자들간에 협동과 경쟁이 이루어지므로 게임 패턴의 다양성(variety)이 거의 무한하다고 할 수 있다. 온라인게임의 두번째 특성은 영속성(persistency)이다. 기존 패키지형 게임은 단기간 내에 게임의 목표를 달성하고 나면 다시 게임을 재시작해야 하는 반복적 특성을 지닌다. 이에 반하여 온라인 게임에서는 게임서버에 의하여 모든 플레이어의 게임 진행상태가 저장, 관리되며 다수의 참여자에 의하여 다중게임이 이루어지는 방식으로 목표달성에 몇년이 걸리거나 거의 무한히 진행될 수도 있다. 또한 기존의 게임에서는 새로운 버전이 출시되어 재설치하기 전에는 게임의 내용을 바꿀 수 없는 반면 온라인 게임은 온라인 상에서 지속적인 업데이트를 통하여 점진적으로 진화하는 게임 콘텐츠를 즐길 수 있는 장점이 있다.

온라인 게임은 영화와 유사한 종합컨텐츠로써 다양한 요소기술이 유기적으로 결합되어야만 제작이 가능하다. 온라인 게임 제작에 필요한 1차적인 기반기술에는 게임 시나리오 창작기술, 게임서버시스템 구현기술, 그래픽스 애니메이션 기술 등이 있으며, 2차적으로 게임의 실감도를 높이기 위하여 물리엔진, 인공지능, 3차원사운드 재현 기술 등이 요구된다.

본 고에서는 온라인게임 기술분야 중 효율적으로 게임공간을 확장하고 대규모 동시참여자를 수용, 관리하는데 필요한 분산게임서버구성기술과 공간관리기술에 초점을 맞추어 기술하고자 한다.

## 2. 온라인 게임의 기술개요

온라인 게임은 2명 이상의 참여자가 통신망상에서 게임의 목적을 달성하기 위하여 상호 협동, 경쟁하는 게임을 의미한다. 동시참여자의 수가 수천명에서 수만명에 이르는 경우 통상적으로 MMPOG(Massively Multi-Player Online Game)라고 부르며 특히, 게임의 장르가 역할수행형(Role-Playing)인 경우 MMORPG(Massively Multi-Player Online Game)라고 부른다.

### 2.1 온라인게임의 개요

최근 온라인게임의 기술의 변화는 다음과 같은 경향을 띄고 있다[1]. 첫째, 표현매체의 발전이다. 텍스트기반의 MUD(Multiple User Dungeon)로부터 출발한 온라인 게임은 2차원 그래픽스를 가미한 MUG(MUD with Graphics)로 발전되었으며, 현재에는 3차원 그래픽스 기술을 바탕으로 하는 Full 3D MMORPG가 주류를 이루고 있다.

두번째 변화는 플랫폼의 다양화이다. 온라인게임은 표 1과 같이 별도의 클라이언트 프로그램과 서버간의 통신으로 진행되는 C-S형 게임, 웹브라우저 환경에서 진행되는 웹보드형 캐주얼게임, 휴대폰 환경에서 수행되는 모바일 게임으로 수행 플랫폼이 다양화되고 있다. 향후에는 게임 실행 플랫폼이 상이한 환경의 참여자들간에도 상호 연동이 가능한 다중플랫폼 온라인게임이 일반화될 것으로 전망된다.

세번째, 게임참여자의 대규모화이다. LAN환경에서 IPX를 사용하여 4명~8명이 참여할 수 있었던 네트워크게임은 수만명 수준의 동시접속자가 가상공간에서 공동으로 게임을 즐기는 형태로 MMORPG로 발전하였다(표 2). 3차원 MMORPG게임인 '리니지 II'는 10만명 이상의 동시접속자수를 기록하며 '리니지'와 더불어 MMORPG시장을 이끌고 있다 [2].

네번째는 게임장르의 다양화이다. 대부분의 MMPOG가 역사나 신화를 배경으로 하는 판타지형 RPG게임이

었으나 최근에는 RTS(Real-Time Strategy), FPS (First-Person Simulation), Action, 스포츠, 레이싱, 무협 등의 장르로 다변화하고 있다.

표 1. 플랫폼에 의한 온라인 게임의 분류

게임수행플랫폼	특징
웹보드형 게임	웹환경에 구축된 게임포탈에서 방의 개념으로 동시참여자의 수를 제한하는 형식으로 진행되는 게임
MMPOG, MMORPG	수백명이상의 대규모 참여자가 가상공간에서 공간적 제약없이 상호작용하는 형식의 게임으로 별도의 클라이언트와 서버 또는 클라이언트와 클라이언트간의 통신으로 진행되는 게임.
모바일 게임	휴대폰이라는 제한된 자원 환경에서 다수 참여자 간에 기존의 온라인 게임과 유사한 내용과 형식으로 진행되는 게임.

표 2 주요 온라인 게임의 동시접속자수 [3]

게임명	장르	제작사	동시 접속자수	특징
Lineage I, II	MMORPG	NCSOFT	122,000	2D, 3D
포트리스2 블루	웹보드 게임	CCR	110,000	캐주얼 게임
뮤(MU) 온라인	MMORPG	웹젠	85,000	3D
라그나로크	MMORPG	그라비티	28,000	3D+2D 카툰렌더링
미르의 전설2/3	MMORPG	액토즈 소프트	12,000	3D, 무협
라그하임	MMORPG	나코 인터랙티브	30,000	3D

## 2.2 네트워크기반 게임의 분류

온라인 게임과 네트워크 게임은 다중사용자가 통신망 상에서 공동으로 게임을 진행한다는 점에서 유사하지만 실제로는 구별되고 있다. 네트워크 게임은 기존 PC게임에 통신기능을 추가하여 16명이내의 소규모 사용자가 LAN환경에서 세션(session)방식으로 진행되는 게임 형식을 말한다. 보통 클라이언트 중에 하나가 로비서버(lobby server) 역할을 담당하고 참여자의 수가 일정수에 도달하면 세션이 개시된다. 반면 온라인 게임은 인터넷 상에서 수백~수천명 이상의 대규모 사용자가 동시에 참여하여 다중으로 게임이 진행되는 형태이다.

온라인게임은 드롭인(drop-in)방식을 지원하여 수시 참여가 가능하고 참여자의 게임진행상태는 서버에 영속적으로 저장된다. MMPOG이라 하면 일반적으로 동시사용자의 수가 수천명 이상인 게임을 의미한다. MMORPG게임의 경우 동시접속자수는 5~10만명 정도이며, 한게임, 넷마블과 같은 웹보드형 게임에서 20만명을 상회하

고 있다 [3]. 대규모 동시참여자들은 게임공간에서 커뮤니티를 형성하고 상호 협동과 경쟁을 바탕으로 게임을 수행함으로써 기존 stand-alone형 게임의 단점이었던 단순한 게임 패턴을 다양화하는데 매우 효과적이라고 할 수 있다. 표 3은 네트워크기반 게임을 특성에 따라 분류한 것이다.

표 3. 네트워크기반 게임의 분류 [3]

분류	특징	사례
머드게임 (1994~)	텍스트를 기반으로 채팅하는 것과 유사한 형식의 온라인 게임	단군의 땅 주라기 공원 (마리텔레콤)
머그게임 (1996~)	머드에 그래픽을 가미한 머드형 온라인게임	바람의나라(넥슨)
소규모 네트워크 게임(1998~)	랜환경에서 16명 이하의 소규모 사용자간의 세션방식으로 이루어지는 형식의 게임, 게임의 내용을 업데이트하기 위해서는 새로운 버전이 출시되어야 함.	스타크래프트 (블리자드) 디아블로II(블리자드)
대규모 온라인 게임(1998~)	16명 이상~수천명. 드롭인 방식으로 참여자가 시간이 제한받지 않고 참가가능. 게임의 내용은 영속적으로 서버에 저장되며, 지속적인 업데이트 가능	리니지(엔씨소프트), 뮤(웹젠), 라그나로크(그라비티), 미르의전설(위메이드)
캐주얼게임, 웹보드게임 (1999~)	웹브라우저 상에서 수행되며 동시게임참여자의 수가 방단위로 제한되는 보드형 게임, 아케이드, 액션, 스포츠 형식의 장르채택	한게임(NHN), 넷마블(플레너스), 포트리스(CCR)

## 3. 분산게임서버 기술

MMORPG에서 분산게임서버 기술이란 동시참여자의 수가 증가하거나 게임공간의 확장되는 경우 발생하는 대량의 데이터를 2대 이상의 분산된 게임서버를 이용하여 병렬처리하는 기술을 의미한다.

### 3.1 게임서버의 기능

온라인 게임에서 게임서버는 표 4와 같이 참여자 관리, 상태정보관리, 게임 시나리오 진행의 기능을 수행한다. 이러한 많은 기능의 수행은 게임서버, 로그인서버, DB서버, 빌링서버 등의 다수 서버간 연동에 의하여 이루어진다. 게임진행과 관련된 기본적인 참여자 관리, 상

표 4 게임서버의 기능

분류	주요 기능
참여자관리	로그인, 로그아웃, 상태정보저장, 타 참여자에게 통지
동적상태정보관리	게임진행과 관련된 참여자의 이동, 레벨, 아이템, 파티설정 등 상태정보관리
공간관리, 서버통신	서버간 통신, 공간관리, 메시지필터링

태정보 저장과 같은 기본적인 기능 이외에 공간분할관리, 메시지필터링 성능은 시스템의 확장성을 결정하는 중요한 기능이다.

고품질의 게임진행을 제공하기 위해서 게임서버는 다음과 같은 요구조건을 만족시켜 주어야 한다.

첫째, 대규모 참여자들을 수용하기 위하여 확장가능한 구조(scalable architecture)를 가져야 한다. 즉, 참여자의 증가와 게임공간의 확장에 대하여 시스템의 성능을 일정하게 유지할 수 있어야 한다. 두번째, 일관성(consistency) 유지의 특성이다. 즉, 동기화되어야 하는 범위내의 모든 참여자들은 원활한 상호작용을 수행하기 위해서 게임공간에 대한 동일한 상태 정보를 가지고 있어야 한다. 모든 참여자는 자신의 관심영역 내에서 발생하는 이벤트 정보를 즉시 수신하여야 한다. 셋째, 최소지연성(latency)을 만족시켜야 한다. 게임의 생명은 응답성이다. 아무리 잘 만들어진 게임이라도 참여자가 조작한 행위에 대하여 피드백이 늦게 전달된다면 게임으로써의 생명을 잃는 것과 같다. 원활한 게임의 진행을 위하여 이벤트 발생으로부터 최대 250ms의 시간지연을 만족시켜주어야 한다 [4].

### 3.2 게임서버의 통신구조

상태 메시지(state update message)는 참여자로 인해 발생한 이벤트를 타 참여자에게 전달하는 수단이다. 상대를 공격하거나 아이템을 습득하는 것과 같이 참여자들이 게임공간에서 발생시킨 이벤트는 패킷(packet)의 형태로 서버를 경유하여 동기화해 주어야 할 클라이언트 그룹에게 전달된다. 표 5는 온라인 게임에서 사용되는 패킷의 종류와 발생요인을 정리한 것이다[5]. 파티(party)란 참여자들이 협동하여 게임을 진행할 목적으로 즉흥적으로 결성된 그룹을 말하며 같은 파티에 속한 멤버는 경험치(experience point) 등을 분배하여 갖게 된다. 상태메시지를 효과적으로 전달하기 위한 서버-클라이언트 통신구조는 표 6과 같이 요약할 수 있다[4].

온라인 게임은 참여자의 참여방식에 따라 모든 플레이어가 게임시작시 동시에 참여하는 세션(session)방식과 게임진행 중에 수시로 출입이 가능한 드롭인(drop-in) 방식으로 구분된다. 세션 방식은 일정수의 플레이어가 입장할 때까지 기다려야 하는 특징이 있는 반면, 드롭인 방식에서는 서버가 지속적으로 참여자 변화로 인한 상태정보를 관리하고 전달해주어야 하는 요구조건이 따른다[4]. 게임공간에서 상대를 만나는 방식에 따라 대기큐에 등록된 후 팀이 구성될 때까지 기다리는 대기큐방식(예, 베틀넷)과 게임을 요청하고 이에 응하는 상대가 있을 경우에 게임이 이루어지는 요청-선택방식(예, 한계

입)이 있다. 리니지와 같은 MMORPG게임은 가상공간을 돌아다니다가 우연히 상대를 만나서 게임이 진행되는 공간 조우방식에 해당된다[6].

표 5 메시지 패킷의 종류[5]

패킷의 종류	패킷 발생 요인
참여자 상태변화패킷	참여자의 로그아웃, 로그인
이동패킷	참여자의 위치이동
행동패킷	조작, 공격명령, 특정스킬사용
게임상태변화패킷	레벨업, 아이템습득, 아이템장착/해제
파티상태패킷	파티설정, 파티해제

표 6 서버-클라이언트간 통신 구조[4]

구분	플레이어 수	동작방식 및 특징
peer-to-peer	최대 16명	client가 메시지 전달 기능 수행, 통신 부담 최소, 확장가능하지 않음
client-server	16~200명	server가 메시지 중계 기능 수행, 서버에서 message filtering 가능 참여자의 수에 따라 서버의 부하가 증가됨, 서버의 다운이 전체 시스템의 다운으로 연결됨
map-based distributed server	200명 이상	가상공간을 region으로 분할하고 각 서버에 하나의 region을 담당, 확장가능, 서버간 통신 부하 발생

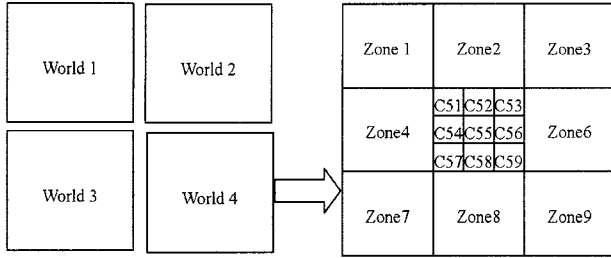
## 4. 게임공간관리기술

### 4.1 공간분할기법

게임공간은 분할된 공간간에 연결되는 방식에 따라 Zoned world와 Seamless world로 구분된다[4]. Zoned world 방식은 전체 게임공간을 상호독립적인 zone으로 구분하여 놓고 각 zone간에는 포탈을 통하여 이동하는 방식이다. 이에 반하여 Seamless world 방식이란 전체 게임공간을 여러 개의 지역으로 분할하고 각 지역에 담당서버를 할당하여 관리하는 방식으로 지역간에 포탈을 경유하지 않고 이동한다. MMORPG는 일반적으로 게임공간의 크기에 제한을 받지 않으면서 게임에서 발생하는 데이터와 메시지 트래픽을 효율적으로 처리하는 구조인 맵기반 분산 서버 구조(map-based distributed server architecture)를 사용된다. 이 방식은 게임공간을 처리가능한 지역단위로 분할하고 각 지역당 담당 서버를 두어 가상공간의 확장성을 높여주는 기법이다. 게임공간을 분할하는 방식에 따라 전체시스템의 효율성이 좌우된다고 할 수 있다.

공간다중분할방식(multi-resolution spatial partitioning)은 온라인게임의 공간을 top-down방향으로 세분화하여 관리하는 방법[7]이다. 전체 게임공간은 상

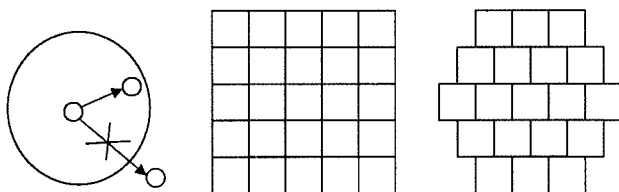
호작용이 불가능하고 서로 독립적인 world의 모임으로 구성된다. 하나의 world는 여러 개의 zone으로 분할되고 각각의 zone은 담당서버에 의하여 관리된다. zone과 zone간의 경계는 시스템만이 알고 있는 정보로 참여자는 인식하지 못한다. zone은 또다시 최소공간단위인 cell로 나누어지며 참여자가 속한 cell과 주변 cell의 집합으로 관심영역(area of interest : AOI)이 정의된다. 참여자의 관심영역인 AOI는 셀을 기반으로 하는 경우와 셀과 무관하게 정의되는 경우도 있다.



Game Environment

그림 1 공간다중분할기법

관심영역이란 참여자가 관심을 가질 것으로 예상하는 주변의 영역을 설정하는 수단으로 발생하는 이벤트의 전파를 관심영역 이내로 제한하여 불필요한 메시지 트래픽을 필터링하는 효과가 있다. 대표적인 AOI의 구성방식으로 Circle-type AOI[8], Cell-based squaring AOI[9], Interleaved-squaring AOI[10]가 있다. Circle-type AOI는 참여자를 중심으로 일정거리 이내의 원형태의 영역을 정의한 것으로 참여자의 움직임에 따라 함께 관심영역도 이동하므로 정확한 관심영역을 유지할 수 있다는 장점이 있는 반면, 참여자가 움직일 때마다 관심영역 안에 포함되는 객체를 업데이트 해주어야 하는 단점이 있다. Cell-based squaring AOI는 공간을 일정크기의 셀로 분할하고 참여자를 둘러싸는 셀들로 관심영역 구성하는 방법이며, 관심영역을 관리하기는 쉬우나 주위의 셀들과 관심영역 중심과의 거리가 일정하지 않은 특징을 갖고 있다. 이러한 단점을 보완한 것이 Interleaved-squaring AOI 방식으로 셀을 엇갈리게 배치하여 관심영역의 중심으로부터의 거리를 균일하게 유지하고자 하였다.



(a) circle-type AOI (b) cell-based squaring AOI (c) interleaved squaring AOI

그림 2 관심영역모델의 종류

## 4.2 경계영역공유기법

앞 절에서 소개한 다중공간분할 방법에서는 가상공간을 여러 개의 작은 영역으로 분할하고 각 영역을 다른 서버가 관리하는 방법이다. 참여자가 두 영역의 경계를 넘어서는 경우 각 영역의 담당서버간의 참여자의 정보를 알려주는 통신 과정이 필요하다. 그림 3에서 만약 2번 참여자의 상태정보를 경계를 넘어서는 순간에 B영역 담당서버로 알려준다면 3번 참여자는 보이지 않던 2번 참여자가 갑자기 앞에 나타나는 현상을 발생시킨다. 무경계 가상공간(seamless boundary virtual environment)을 구현하기 위해서는 두 영역간 경계 주위에 공유영역을 두어 갑작스럽게 참여자가 나타나는 현상을 제거하는 것이 바람직하다. 경계영역공유기법[10]이란 두 영역의 경계를 중심으로 참여자의 가시권(관심영역) 이내의 지역에 존재하는 참여자의 상태정보를 양쪽 서버에서 함께 공유하는 방법으로 앞서 언급한 갑작스러운 출현을 방지할 수 있다. 다시 말하여 2번 참여자가 공유영역에 진입하는 순간 B지역 담당서버에 상태정보를 전송하고 B지역 서버는 자신의 영역 내의 참여자중 관련자에게 이 사실을 통지함으로써 3번 참여자는 2번 참여자가 가시권 내에 들어오는 순간 미리 인식할 수 있다. 이 방법은 단점으로는 공유영역 내에 존재하는 참여자의 상태정보가 그 영역을 공유하는 서버의 수에 비례하여 중복되어 관리되어야 한다는 점이다.

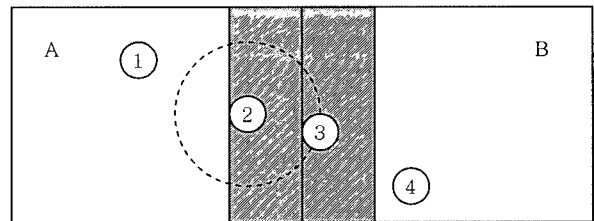
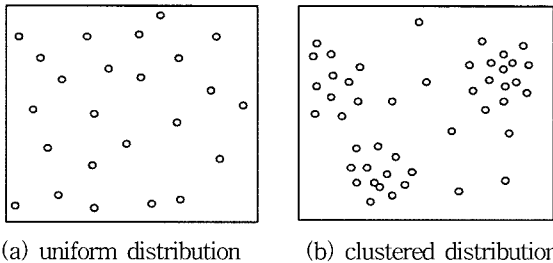


그림 3 경계공유영역

## 4.3 분산서버의 부하균등화

게임공간에서 참여자들은 전체 공간에 균일하게 분포하는 경우보다는 특정지역에 편중되어 분포하는 경우가 더 일반적이다. 맵기반 분산서버방식에서 그림 4와 같이 참여자의 밀집도가 높은 지역을 담당하는 서버에게는 메시지 트래픽 처리 부하가 증가되며, 상대적으로 밀집도가 낮은 지역의 서버는 처리할 데이터가 적은 부하불균형(workload unbalance)의 문제가 발생한다. 이러한 분산 서버의 처리부하를 균등화(load balancing)하여 시스템을 안정화하는 문제는 최근 온라인게임분야의 주요 이슈가 되고 있다 [12]. 맵기반 분산서버방식에서 서버 부하의 불균형은 불가피하며 효과적으로 균등화하기

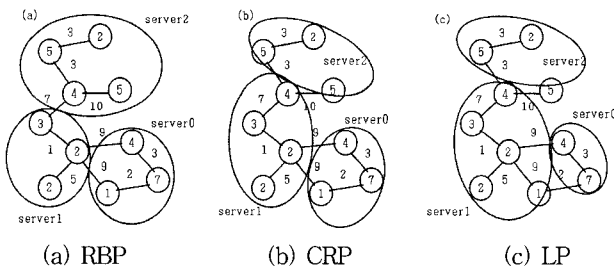
위하여 새로운 기준을 적용하여야 한다. 분산서버간 부하균등화 문제는 최적의 분할을 탐색하기까지 너무 많은 경우의 수가 발생하는 NP-complete problem로 알려져 있다 [13].



(a) uniform distribution (b) clustered distribution

그림 4 참여자 분포도

Lui와 Chen이 제안한 Linear Optimization Technique(LOT)기법[14]은 그래프 이론(graph theory)을 서버 부하 균등화 문제에 적용한 것으로 가장 효율적인 방법으로 평가되고 있다[13]. LOT기법은 각 참여자를 노드로 표현하고 AOI의 충돌이 일어난 참여자들간에는 노드를 edge로 연결하고 떨어진 거리를 edge의 weight로 표시한다. 또한 각 참여자로 인해 발생할 서버의 처리 부하(관심영역 내의 참여자의 수)를 노드의 label로 표시한다. LOT알고리즘은 이러한 형태로 표현된 그래프를 대상으로 Recursive Bisection Procedure(RBP), Layering Partitioning Procedure(LP), Communication Refinement Partitioning(CRP)의 3단계로 구성된 그래프 분할 알고리즘을 통하여 부하의 불균형 수치가 지정된 임계값(threshold) 이하에 도달할 때까지 그래프를 분산서버의 갯수로 분할하는 과정을 반복한다 [14].



(a) RBP (b) CRP (c) LP

그림 5 LOT 방법에 의한 가상공간의 분할

## 5. MMORPG 게임엔진 기술

게임 엔진이란 온라인게임의 수행을 위하여 필요한 서버와 클라이언트 소프트웨어들의 집합을 의미한다. 게임엔진에는 수행기능별로 그래픽스엔진, 물리엔진, 인공지능 등의 서브엔진모듈(sub-engine module)이 포함된다[3]. 게임서버는 통신 네트워크에서 다수의 사용자가 공동으로 게임을 플레이할 수 있도록 통신환경을 제

공하는 것이 게임 서버의 주기능이다. 게임서버는 참여자 관리, 게임상태 저장 및 관리, 게임공간관리, 이벤트 전송, 메시지 필터링과 같은 시스템 전반에 관한 중요한 기능을 수행한다. 게임서버의 성능이 온라인게임의 확장성과 응답성을 결정한다. 그래픽 엔진은 그래픽엔진은 3차원 렌더링과 애니메이션을 담당하는 모듈로서, 화면을 구성하는 물체 및 효과를 실시간으로 처리하여 사실적이고 생생한 이미지를 재현하는 것이 주된 기능이다. 인공지능 엔진은 인공지능은 프로그램을 통하여 NPC(Non-Player Character)나 상대(적)의 행동 패턴을 다양하게 시뮬레이션하기 위한 기술이다. 지능적인 상대를 요구하는 사용자의 욕구가 커짐에 따라 정교한 인공지능 기술의 개발이 활발하다. 물리엔진은 게임 공간에 등장하는 캐릭터 혹은 오브젝트의 움직임을 사실적으로 표현하기 위하여 자연법칙에 기반한 재현방법을 제공한다. 사운드 엔진은 3D사운드 엔진이란 3차원의 현장감 있는 음향을 구현하는 기술로 실시간으로 객체의 위치를 파악해서 음을 생성하여 제공하는 입체 음향을 말한다.

표 7은 주요 게임엔진의 특성을 정리한 것이다[1]. Quake엔진은 네트워크 기능이 탑재된 3차원 게임엔진으로 클라이언트-서버의 기능을 구현하여 단독으로 게임을 플레이할 수 있도록 구현한 엔진이다. 서버는 인공지능 기능, 캐릭터의 자유로운 동작 기능, 충돌과 물리적인 현상 기능을 Quake-C라는 인터프리터 언어를 개발하여 게임 콘텐츠의 개발을 용이하다. Unreal엔진은 Quake 게임엔진을 한 단계 발전시킨 엔진으로 Unreal Script라는 스크립트 언어를 통한 객체지향형 상속기능을 지원하여 새로운 특성을 지닌 캐릭터나 몬스터를 쉽게 탄생시킬 수 있다[9].

Infinity 게임엔진은 1998년 개발한 RPG 전용 게임엔진으로, Buldur's Gate 시리즈 등의 게임제작에 사용되었다. 이 엔진은 인공지능을 지원하는 AICompile 스크립트 언어를 제공한다. 특히 AICompile 스크립트는 규칙기반시스템(Rule based System) 구조를 채용하여 특정상황에서 캐릭터의 동작을 쉽게 표현할 수 있으며, RPG게임에서 자주 등장하는 NPC와의 대화 표현도 용이하게 구현할 수 있다. AGE OF EMPIRE II 게임엔진은 실시간 전략게임(Real Time Strategy)으로서 인공지능 기능을 처리하기 위하여 스크립트 언어를 사용한다. 전략게임의 경우에 공격, 건설, 점령과 같은 다양한 목표설정을 이해하고 처리하기 위하여 인공지능 기능이 요구된다. 또한 전략게임의 경우에 지형을 분석하여 유닛을 이동시키는 것은 매우 중요한 기능중의 하나이다.

각 게임 개발사마다 자사의 게임만을 위한 독자적인

게임엔진을 제작하여 왔다. 게임엔진은 참여자 관리, 이벤트 처리, 그래픽스 처리와 같은 유사한 기능을 포함함에도 불구하고 중복적인 개발로 인하여 많은 시간과 비용이 낭비되었다고 할 수 있다. 이러한 문제점의 해결책은 범용게임엔진(general-purpose game engine)의 개발이다. 범용게임엔진이란 참여자관리, 클라이언트-서버 통신 등의 기본적인 기능을 포함하고 개발적인 응용게임을 개발하는데 필요한 API(application programming interface)를 제공하는 게임엔진을 말한다. 범용게임엔진의 개발은 게임엔진 소프트웨어의 재활용을 통하여 중복투자비용 절감, 게임개발기간의 단축과 같은 장점뿐만 아니라, 게임엔진 개발에 들어가는 비용과 노력을 게임 시나리오 구성과 같은 콘텐츠 제작에 집중함으로써 양질을 게임 콘텐츠를 제작하는 효과를 거둘 수 있다 [11].

표 7 게임엔진의 기능 비교 [1]

게임 엔진	상용게임	장르	동시접속자수	제작사
Quake Engine	Quake, Half-life	Action	~10	ID Software
Unreal Engine	Unreal, Qeus-Ex	Action	~10	Epic Mega Games
Turbine Engine	Asheron's Call	MMORPG	~1000	MGS
NetImmense	Dark Ages of Camelot	Action, MMORPG	~1000	NDL
뮤	MU	MMORPG	~1000	웹젠

## 6. 결 론

게임서버개발기술은 제한된 통신대역폭을 가지고 얼마나 많은 참여자에게 어떻게 최소의 응답성을 제공할 것인가하는 문제로 요약할 수 있다. 현재 서버기술로는 10만명이상의 동시접속자를 지원하기 어렵다고 알려져 있으나 앞으로의 온라인게임에서 100만명 수준으로 접속자가 증가한다고 볼 때 확장가능한 게임서버의 개발은 매우 중요한 이슈라고 할 수 있다. 현재의 MMORPG는 단순히 게임을 목적으로 하는 엔터테인먼트분야에 국한되지만 향후에는 cyber-community와 e-business, e-learning등의 정보시스템들과 통합되어 edutainment 또는 infotainment형태로 발전할 것으로 예상된다. 따라서 가상공간에서 동시접속자의 수는 현재와 비교할 수 없을 정도로 증가할 것이며 이를 수용할 수 있는 확장가능한 서버의 개발은 시급하다.

미래의 게임공간에서는 참여자가 zone간의 경계를 넘어서는 경우 서버간 통신으로 인한 응답 지연(response delay)을 느낄 수 없는 무제한적인 공간관리기술의 개

발과 함께, 다중참여자들이 수시로 게임공간에 출입이 가능한 Drop-in방법이 일반화되면서 게임서버에서 영속적으로 참여자의 게임상태를 유지관리하기 위한 방안이 수립되어야 한다.

국내의 온라인 게임기술의 세계적인 우위에도 불구하고 폐쇄적인 개발구조로 인해 중복 투자 및 과잉경쟁이 발생하고 있으며 이것은 국내 게임분야의 기술적, 산업적 발전을 저해하는 요인이 되고 있다. 향후 게임시장의 성패가 시나리오 구성, 애니메이션과 같은 콘텐츠 개발에 달려있다고 볼 때 국내 게임개발사간의 상호 공동협력력이 매우 절실한 시점이라고 할 수 있다.

게임진행관리	Session	Drop-in		
게임공간관리	Zoned World	Seamless World	Unlimited World	
다중참여기술	10,000	100,000	1,000,000	
아키텍처	P2P	C-S	Distributed	
	2000	2002	2004	2006 2008

그림 6 분산게임서버 요소기술의 발전방향[3]

## 참고문헌

- [1] 이현주, 김현빈, "아리아온라인 : Dream3D를 이용한 온라인게임", 멀티미디어학회논문지, 제7권 4호, 2004, pp. 532-541.
- [2] Lineage web site, <http://www.lineage.com>.
- [3] 2004 대한민국 게임백서, 한국게임산업개발원, 2004.
- [4] 양광호, 심광현, 고동일, 박일규, 김종성, "온라인 게임 서버의 기술동향", 전자통신 동향분석, 제16권 4호, 2001, 8호.
- [5] 남재욱, 온라인게임서버프로그래밍, 한빛미디어, 2004.
- [6] Ben Calica, Multi-player lobbying, Gamasutra, Sep. 4, 1998
- [7] B. Ng, A. Si, R. W.H. Lau, F. W.B. Li, A Multi-Server Architecture for Distributed Virtual Walkthrough, Proceedings of the ACM symposium on Virtual Reality Software and Technology, November 2002, pp.163-170.
- [8] C. M. Greenhalgh, and S. D. Benford, MASSIVE: A Distributed Virtual Reality System Incorporating Spatial Trading, Pro-

- ceedings of 15th International Conference on Distributed Computing Systems, Los Alamitos CA, ACM Press, 1995, pp. 27-34.
- [9] 김성환, 박태준, "게임컨텐츠의 기획과 게임엔진 기술", 한국멀티미디어학회 논문지, 제8권 제1호 2004, pp. 1-15.
- [10] J. Huang, Y. Du, C. Wang, Design of the Server cluster to support avatar migration, Proceedings of the IEEE Virtual Reality 2003.
- [11] 이현주, 김준애, 임충규, 김현빈, "온라인 3D 게임 엔진 표준화", 한국정보처리학회지, 2002년 5월 호.
- [12] C.S. John, M. Lui, F. Chan, An Efficient Partitioning Algorithm for Distributed Virtual Environment Systems, IEEE Transactions on Parallel and Distributed Systems, Vol. 13, No. 3, March 2002, pp. 193-211.
- [13] P. Morillo, J. M. Orduna, M. Fernandez, A Comparison Study of Evolutive Algorithms for Solving the Partitioning Problem in Distributed Virtual Environment Systems, Parallel Computing Vol. 30, 2004, pp. 585-610.
- [14] J.C.S. Lui, M.F. Chan, K.Y. Oldfield, Dynamic Partitioning for a Distributed Virtual Environment, Department of Computer Science, Chinese University of Hong Kong, 1998.

---

유 석 종



1994. 2 연세대학교 컴퓨터과학과(학사)  
 1996. 2 연세대학교 대학원 컴퓨터과학과(석사)  
 2001. 2 연세대학교 대학원 컴퓨터과학과(박사)  
 2001. 6~2002. 12 University of Ottawa, Postdoctoral fellow  
 2003. 1~2005. 2 국립한밭대학교 정보통신컴퓨터공학부 전임강사

2005. 3~현재 숙명여자대학교 정보과학부 조교수  
 관심분야: 컴퓨터그래픽스, 가상현실, 온라인게임  
 E-mail : yusjong@sookmyung.ac.kr

---